AD 296 590

SDC

TM-602/400/00

The Dependency of a Simulation Language On a Theory of Systems

28 January 1963

TM-602/400/00

TECHNICAL MEMORANDUM

(TM Series)

This document was produced in connection with a research project sponsored by SDC's independent research program.

The Dependency of a Simulation Language On a Theory of Systems

by

Michael R. Lackner

January 28, 1963

SYSTEM DEVELOPMENT CORPORATION
2500 COLORADO AVE.
SANTA MONICA
CALIFORNIA

A-2345 4/62

THE DEPENDENCY OF A SIMULATION LANGUAGE ON A THEORY OF SYSTEMS

This paper discusses the relationship of a simulation language to a theory of systems. Realization of the great potential of digital simulation seems to await development of a language capable of facile expression of a wide variety of systems. A language adequate for this task, however, must stem from a general theory of systems. For it is such a theory which will enable deduction of a system's behavior from a description of the system. Specification of the language must be dictated by the requirements of the theory; language must be capable of theoretically coherent expression before catering to its user's desire for easy expression.

In the development of digital simulation systems, much more attention has been paid to language than to theory. Because any programming language can be used to code a simulation model, the view has developed that more economical production of simulation models can be achieved by simply refining existing languages. This view can lead to the unfortunate result that production of specious computer programs becomes easy while the fundamental problem of producing valid simulation models remains as difficult as before. A language lacking firm roots in theory can only facilitate incoherent, inconsistent or ambiguous expression.

Some terms which are used throughout this paper are defined as follows:

"Model": A representation of an aggregate of phenomena.

"System": A body of phenomena forming an organic whole.

"Object system": A system to be modeled; the system under study.

"Simulation": Dynamic representation of an object system. A simulation model of an object sytem is itself a system whose behavior may be interpreted according to a rule which renders a description appropriate to the behavior of the object system.

Models and Simulation

All simulation involves the use of models. A system to be simulated is represented by a model. Prediction of certain aspects of system performance is the usual purpose of simulation. The particular purpose of a simulation determines the detailed characteristics of a model, but its more general characteristics are determined by the form of the model. Geometrical constructions, for example, will vary with the spatial dimensions of the objects they represent, but all such constructions possess certain common characteristics. Most important, they are all in geometrical form, and permit the exercise of geometrical theory. Determinations of qualities and measures of the models may be made by applying the theory, and these qualities and measures may hold true of the modeled objects.

But models are caricatures in the sense that all attributes of an object system are not faithfully represented in the model; some may be grossly exaggerated relative to others, and some entirely omitted. What a modeler considers relevant to the purpose of the model is included; what he considers irrelevant is excluded. For this reason a model will serve certain purposes, but be entirely inadequate for others. For instance, a motorist's road map may serve to guide him from city to city, but be almost useless to a pilot, geologist or weather forecaster interested in the same geographical area. It may become useless to the motorist when he attempts to locate a street within a city.

What is represented in a model is a function of the model's purpose; how it is represented is a function of the form chosen for its implementation. The choice of form is critical. The graphic form of the motorist's map permits his reading it, while another form of representing the same information, say magnetic tape, might preclude his reading it. Just as the physical form determines who may read a map, the logical form of a model determines what theory will recognize the model as an instance of its subject matter, and so be capable of dealing with it.

Simulation Languages

A simulation language is a language designed to facilitate the construction of a digital simulation model of an object system. Such a model consists of data descriptions of various phenomena and logical operations upon the data. The operations represent dynamic interrelationships among phenomena represented by the data descriptions.

The construction of digital simulation models is a laborious and expensive undertaking. The native language of a computer is its basic instruction set; thousands of basic instructions are required in even fairly simple models. Languages have been developed for easier and more efficient coding of procedures likely to be used in certain computer applications. Recently, languages peculiarly suited to the generation of simulation models have been developed. This has proven difficult work, because procedures peculiar to digital simulation are not as well established as are those in some other areas. In mathematics, procedures for solving equations existed long before digital computers; in data processing, procedures for sorting and merging records have been implemented by machine for 50 years. Indeed, analog computers have been implementing analog simulations for a very long time. Digital simulation of large systems, however, has existed only since high-speed, electronic digital computers, and its theory and procedures are in an early stage of development.

There is some commonality among the various approaches so far taken toward the problem of constructing a general simulation capability. Each has included the identification of a general form a model will possess, and the construction of a language for expressing examples of that form. Each system has included a translator or interpreter of the language. Several include a skeletal program which becomes part of each model and manipulates the model to reflect the passage of time.

The computer programs of available simulation systems differ greatly in their structure. The really significant differences, however, among such systems as Dynamo, Forrester (1958); SIMSCRIPT, Markowitz, Hausner and Karr (1962); General Purpose System Simulator, Gordon (1961); General Simulation Program, Tocher and Owen (1960); Control and Simulation Language, Laski and Buxton (1962); and SIMPAC, Lackner (1962), are differences among the theories of systems to which a modeler using one of these tools must subscribe. Unfortunately, the theories are precisely described only by the computer programs which manipulate the models constructed within these systems, although features of the theories are suggested by the languages.

The programs, nevertheless, determine when, where and what changes will be impressed upon the state of the model they manipulate. These determinations are made in accordance with a decision procedure. These simulation systems prescribe a method of producing a model and supply a general procedure for deducing successive states of the model. Such a scheme confesses the presence of a general theory of systems, however crude or undeveloped it may be.

In one sense, each simulation model of an object system expresses a theory of that system; the features of the object system which bear some functional relationship to the measurements that the model is to furnish are identified, and the functional relationships described. But the kinds of features which may be contemplated in any system, and the kinds of functional relationships which may hold, are determined by the general theory of systems on which the whole simulation system is based. Only by appeal to this theory may a modeler say "this is a model of object system x" and "this is how the model behaves." Only by such appeal may he explain why it is a model, and why the model so behaves. (It does not follow that the modeler necessarily knows or will admit has included a translator or interpreter of the language. Several include a skeletal program which becomes part of each model and manipulates the model to reflect the passage of time.

The computer programs of available simulation systems differ greatly in their structure. The really significant differences, however, among such systems as Dynamo, Forrester (1958); SIMSCRIPT, Markowitz, Hausner and Karr (1962); General Purpose System Simulator, Gordon (1961); General Simulation Program, Tocher and Owen (1960); Control and Simulation Language, Laski and Buxton (1962); and SIMPAC, Lackner (1962), are differences among the theories of systems to which a modeler using one of these tools must subscribe. Unfortunately, the theories are precisely described only by the computer programs which manipulate the models constructed within these systems, although features of the theories are suggested by the languages.

The programs, nevertheless, determine when, where and what changes will be impressed upon the state of the model they manipulate. These determinations are made in accordance with a decision procedure. These simulation systems prescribe a method of producing a model and supply a general procedure for deducing successive states of the model. Such a scheme confesses the presence of a general theory of systems, however crude or undeveloped it may be.

In one sense, each simulation model of an object system expresses a theory of that system; the features of the object system which bear some functional relationship to the measurements that the model is to furnish are identified, and the functional relationships described. But the kinds of features which may be contemplated in any system, and the kinds of functional relationships which may hold, are determined by the general theory of systems on which the whole simulation system is based. Only by appeal to this theory may a modeler say "this is a model of object system x" and "this is how the model behaves." Only by such appeal may he explain why it is a model, and why the model so behaves. (It does not follow that the modeler necessarily knows or will admit that this is the case, or will even recognize its importance.) It is according to a theory, simple or complex, spoken or not, that a model is operated, and a model is expressed in accordance with the requirements of that theory.

That the development of a simulation language requires the development of a general systems theory is only natural. A language is a means of expressing thought. The problem in developing a simulation language is the problem of thinking about what is to be simulated. If systems are to be simulated, then a language must be capable of expressing thought about systems, and an understanding of how systems are to be thought of is necessary. Max Black (1962), discussing "the _existential_ _use_ _of_ _models_" as "characteristic of the practice of the great theorists in physics," says "the heart of the method consists in _talking_ in a certain way."[1]

The final expression of a digital simulation model is computer code--algorithms and data--but categorization of system elements as algorithms or data does not produce a useful model. A more restrictive set of categories, whose members fall in the earlier two, is necessary to establishing a general approach to modeling systems.

Such a restrictive set of categories is identified with a _Weltansicht_, a special view or apprehension of reality as a whole, which a modeler adopts when contemplating an object system. A modeler looks at the system in a certain way. Adoption of a Weltansicht includes the assumption of a category set and rules governing their relationships. Certain kinds of things are contemplated, and an orderly scheme of relationships among these kinds of things is assumed.

If "theory" is understood as "a body of general principles offered to explain phenomena," it is apparent that a theory both gives rise to a Weltansicht and depends upon a Weltansicht for its application. To apply geometry, for

---

[1]Italics in original.

example, one must view phenomena as assemblages of entities falling in the category system of geometry. Conversely, such a geometrical Weltansicht is justified only by appeal to geometrical theory.

A language capable of expressing a model must be capable of generating expressions which, taken as a body, comprise a model in a particular form. Ideally, any instance of the form should be expressible in the language. Any hypothesis or observation consonant with the Weltansicht should be expressible. The character of a language for expressing simulation models, the subject matter of a theory of systems, must be determined by the theory.

What has happened in the development of simulation languages is this: the orderly scheme of relationships has been expressed in computer code, the programs which are part of every model. A modeler using the language understands that this scheme is supplied; he does not alter the scheme in using the language; he describes system elements in accordance with the category system, which is also invariant. The simulation language is a part of a simulation system which is based on a theory of systems, but the theory has been neither elucidated nor developed.

Development of a Theory of Systems

Systems exhibit complex behavior. A time series of vector descriptions of an object system will contain, besides variable values of vector dimensions, a variable number of dimensions. Even if the sets of dimensions making up the time series have no common members, historic dependency relationships justify considering the subject of the time series a single system.

Digital simulation techniques have been largely confined to dealing with systems whose components remain the same, however drastic their changes of state. But development of the embryonic systems theory underlying current simulation capability will extend the useful applicability of simulation to systems whose components come into existence, experience several states and disappear. Furthermore, development of system theory should make possible deductions of interesting measures and properties of simulation models which are not dependent on empirical observation of the models' behavior. These latter deductions may be as readily automated as the deduction of sequential states.

Simulation has always dealt with system change. Digital simulation is harnessing the power of computers to deal with change, and development of its theory will enable simulation techniques to deal with change wherever it manifests itself, in the birth, development, behavior or disappearance of systems. Language for the precise definition of models and their behavior will develop with theory.

There is little discernible agreement among those interested in the development of systems theory, but the inviting properties of digital computers may well shape the theory or act as catalyst to contending opinions.

Lektorsky and Sadovsky (1960), in criticism of Bertalanffy's general systems theory, say, "the specific peculiarities of self-developing complex systems of connections and processes are determined by their intertwining, mutual influences, transformations, etc. In other words, for understanding a system it is necessary to clarify the subordination of the elements to the whole, and this is possible only in the framework of a historical approach to system analysis, i.e., through understanding its given condition as having developed and continuing to develop historically."

A theory of systems must dispose of the problems associated with historical development as well as those encountered in stable operation.

Bernard Bosanquet (1920) might have been discussing simulation, and certainly indicated the requirements to be met by a simulation language and its underlying theory, when he discussed systematic inference: "Make a supposition, as complex as you please; say, consisting in the total rules of a game like chess or noughts and crosses. Put into it everything you think necessary to determine the consequences you mean to draw. So far, of course, you have no affirmation, you have only a very complex antecedent of a hypothetical judgment without any consequent. So long as you are merely supposing, the data or contents you suppose, one might say, lie dead side by side. They do not combine or affirm anything about anything; they do not modify or confirm one another or exclude one another or the consequences of one another.

"But now make a judgment, ... affirm consequential bearings of one supposed element on another, ... you have infused the life of reality into your suppositions. ... the train of consequences begins to affirm itself."

A simulation language must serve for stating such suppositions, and also serve for stating the train of consequences. The language must express the subject matter of the theory; the theory must be capable of recognizing the suppositions and drawing the consequences.

BIBLIOGRAPHY

Black, M., 1962, Models and Metaphors, Cornell University Press, 1962.

Bosanquet, B., 1920, Implication and Linear Inference, MacMillan and Company, Ltd., 1920.

Forrester, J. W., 1958, Industrial Dynamics - A Major Breakthrough for Decision Makers. Harvard Business Review, July - August 1958.

Gordon, Geoffrey, 1961, A General Purpose Systems Simulation Program. Proceedings 1961 Eastern Joint Computer Conference, December, 1961.

Lackner, M. R., 1962, Toward a General Simulation Capability. Proceedings 1962 Spring Joint Computer Conference, May, 1962.

Laski, J. C. and Buxton, J. N., 1962, C. S. L. Control and Simulation Language. Esso Petroleum Company, Ltd. and IBM United Kingdom, Ltd.

Lektorsky, V. A. and Sadovsky, V. N., 1960, On Principles of System Research (related to L. Bertalanffy's General System Theory), translation by A. Rapoport in General Systems, 5, 1960.

Markowitz, H. M., Hausner, B. and Karr, H. W., 1962, SIMSCRIPT: A Simulation Programming Language. The RAND Corporation Memorandum RM-3310-PR, November, 1962.

Tocher, K. D. and Owen, D. G., 1960, The Automatic Programming of Simulation. Proceedings of the Second International Conference on Operational Research, English Universities Press, 1960.

UNCLASSIFIED

System Development Corporation,
Santa Monica, California
THE DEPENDENCY OF A SIMULATION
LANGUAGE ON A THEORY OF SYSTEMS.
Scientific rept., TM-602/400/00, by
M. R. Lackner. 28 January 1963,
9p., 9 refs.

Unclassified report

DESCRIPTORS: Data Processing Systems.
Simulation.

Discusses the relationship of a simulation language to a theory of systems. States that in the

UNCLASSIFIED

---

development of digital simulation systems, more attention has been paid to language than to theory. Reports that because any programming language can be used to code a simulation model, the view has developed that more economical production of simulation models can be achieved by simply refining existing languages. States that this leads to the unfortunate result that production of specious computer programs becomes easy while the fundamental problem of producing valid simulation models remains difficult. Concludes that the language must express the subject matter of the theory; the theory must be capable of recognizing the suppositions and drawing the consequences.

UNCLASSIFIED

UNCLASSIFIED